PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 :
G06F 17/60

A1

(11) International Publication Number: WO 99/62018

(43) International Publication Date: 2 December 1999 (02.12.99)

(21) International Application Number: PCT/US99/11926

(22) International Filing Date: 28 May 1999 (28.05.99)

(30) Priority Data:
60/086,994 28 May 1998 (28.05.98) US

(71) Applicant: I2 TECHNOLOGIES, INC. [US/US]; Suite 1600, 909 East Las Colinas Boulevard, Irving, TX 75039 (US).

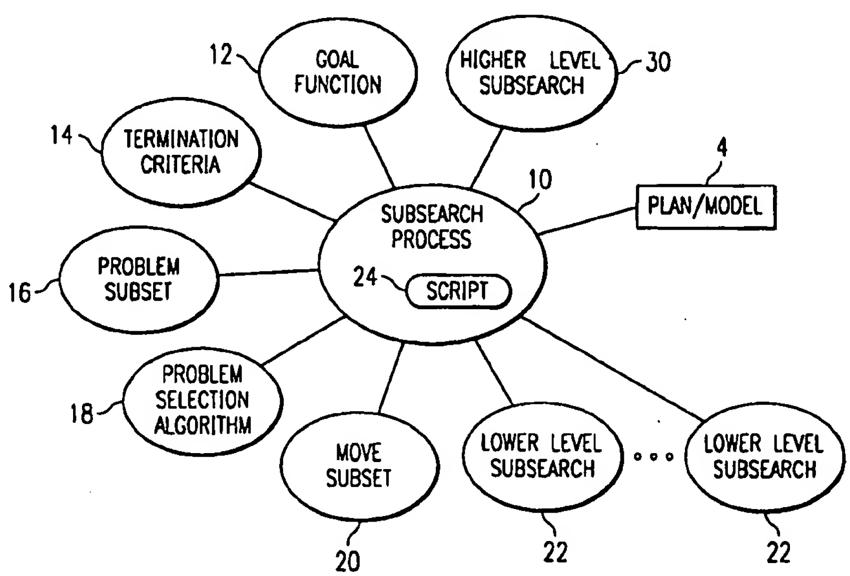
(72) Inventors: CRAWFORD, James, M., Jr.; 2924 Hugo Court, Flower Mound, TX (US). KENNEDY, Brian, M.; 136 Rustic Meadow Way, Coppell, TX 75019 (US). LIN, Tiaohua; 2708 Lake Breeze Lane, Flower Mound, TX 75028 (US). VENKATASUBRAMANYAN, Narayan; 1321 Coral Drive, Coppell, TX 75019 (US). KUNCHITAHAPATHAM, Arun; 4279 N. O'Conner Boulevard, Irving, TX 75062 (US). ZEITHAMMER, Karel; Vinkenlaan 20, B-3080 Tervuren (BE).

(74) Agent: KENNERLY, Christopher, W.; Baker & Botts, L.L.P., 2001 Ross Avenue, Dallas, TX 75201-2980 (US).

(81) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

**Published**
*With international search report.*

(54) Title: COMPUTER IMPLEMENTED SYSTEM AND METHOD FOR HIGH LEVEL CONTROLLED SEARCHING THROUGH A PROBLEM SPACE




(57) Abstract

A system and method for searching a problem space. In an exemplary application, the search constructs a plan associated with a business enterprise and the problem space in a model of the enterprise. The search is divided into subsearches. Each subsearch is performed by a subsearch process having its own goal data, termination data, problem data, problem selection process, move process, and script. The script of a subsearch determines whether the current state of the search shall be advanced by a move of that subsearch or by another subsearch.

## *FOR THE PURPOSES OF INFORMATION ONLY*

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| AL | Albania | ES | Spain | LS | Lesotho | SI | Slovenia |
| AM | Armenia | FI | Finland | LT | Lithuania | SK | Slovakia |
| AT | Austria | FR | France | LU | Luxembourg | SN | Senegal |
| AU | Australia | GA | Gabon | LV | Latvia | SZ | Swaziland |
| AZ | Azerbaijan | GB | United Kingdom | MC | Monaco | TD | Chad |
| BA | Bósnia and Herzegovina | GE | Georgia | MD | Republic of Moldova | TG | Togo |
| BB | Barbados | GH | Ghana | MG | Madagascar | TJ | Tajikistan |
| BE | Belgium | GN | Guinea | MK | The former Yugoslav | TM | Turkmenistan |
| BF | Burkina Faso | GR | Greece | | Republic of Macedonia | TR | Turkey |
| BG | Bulgaria | HU | Hungary | ML | Mali | TT | Trinidad and Tobago |
| BJ | Benin | IE | Ireland | MN | Mongolia | UA | Ukraine |
| BR | Brazil | IL | Israel | MR | Mauritania | UG | Uganda |
| BY | Belarus | IS | Iceland | MW | Malawi | US | United States of America |
| CA | Canada | IT | Italy | MX | Mexico | UZ | Uzbekistan |
| CF | Central African Republic | JP | Japan | NE | Niger | VN | Viet Nam |
| CG | Congo | KE | Kenya | NL | Netherlands | YU | Yugoslavia |
| CH | Switzerland | KG | Kyrgyzstan | NO | Norway | ZW | Zimbabwe |
| CI | Côte d'Ivoire | KP | Democratic People's | NZ | New Zealand | | |
| CM | Cameroon | | Republic of Korea | PL | Poland | | |
| CN | China | KR | Republic of Korea | PT | Portugal | | |
| CU | Cuba | KZ | Kazakstan | RO | Romania | | |
| CZ | Czech Republic | LC | Saint Lucia | RU | Russian Federation | | |
| DE | Germany | LI | Liechtenstein | SD | Sudan | | |
| DK | Denmark | LK | Sri Lanka | SE | Sweden | | |
| EE | Estonia | LR | Liberia | SG | Singapore | | |

# COMPUTER IMPLEMENTED SYSTEM AND METHOD FOR HIGH LEVEL CONTROLLED SEARCHING THROUGH A PROBLEM SPACE

## TECHNICAL FIELD OF THE INVENTION

This invention relates in general to the fields of supply chain management, and single- and multi-enterprise planning. More particularly, the present invention relates to a computer implemented system and method for high level controlled searching through a problem space.

## BACKGROUND OF THE INVENTION

Computer implemented planning systems are widely used for factory, enterprise and supply chain planning functions. In general, the systems model the manufacturing environment and provide plans for producing items to fulfill consumer demand within the constraints of the environment.

A classic artificial intelligence search can include elements such as: an initial state, a goal state, a termination criteria, and a set of moves between states of the search space. For example, in a "blocks world" problem space, an artificial intelligence search engine could have an initial state of a red block and blue block on a table and a yellow block on top of the blue block. A goal state for such an engine could be the goal of building a planned sequence of moves which place the red block on top of the blue block. The termination criteria could be to achieve the goal within 10 minutes or quit. The set of moves could comprise: moving an uncovered block onto the table and

moving an uncovered block from the table onto another uncovered block. An artificial intelligence search process, then, could compute a plan for the required moves to achieve the goal. When applied to planning problems, the number of "blocks" greatly increases, and the initial and goal states become more complex.

With more "blocks" and/or more complex initial and goal states, this type of search can be computationally challenging. Further complexity can be added to a system when the types and numbers of moves grow.

SUMMARY OF THE INVENTION

In accordance with the present invention, a computer implemented system and method for high level controlled searching through a problem space is disclosed that provides significant advantages over previously developed searching mechanisms.

According to one aspect of the present invention, the system can include a plurality of subsearch processes, wherein each subsearch process can comprise: a goal function, a termination criteria, a problem subset, a problem selection algorithm, a move subset, a lower subsearch set, and a script. The problem subset can be an identified flaw, infeasibility, or suboptimal condition in a given system. The problem selection algorithm is operable to select a targeted problem from the problem subset. The move subset is a set of functions or operations operable to optimize the current subsearch. The lower subsearch set is a set of one or more "downstream" subsearches. The script is operable to specify or determine the appropriate moves or subsearches to execute.

It is a technical advantage that the invention can allow convenient expression of complex divide-and-conquer type searches.

It is another technical advantage that the invention allows for combined automated and manual searches, as well as potentially other types of searches.

Other technical advantages should be apparent to one of ordinary skill in the art in view of the specification, claims, and drawings.

## BRIEF DESCRIPTION OF THE DRAWINGS

A more complete understanding of the present invention and advantages thereof may be acquired by referring to the following description taken in conjunction with the accompanying drawings, in which like reference numbers indicate like features, and wherein:

FIGURE 1 is a block diagram of a planning system that provides a mechanism for high level controlled searching through a problem space; and FIGURE 2 is a block diagram of one embodiment of a subsearch process for high level controlled searching through a problem space according to the present invention.

## DETAILED DESCRIPTION OF THE INVENTION

FIGURE 1 is a block diagram of a planning system, indicated generally at 2, that provides a mechanism for high level controlled searching through a problem space according to the present invention. Planning system 2 can be implemented on a computer system having typical computer components such as a processor, memory, storage devices, etc. In the embodiment of FIGURE 1, planning system 2 executes software that implements a planning engine 3 in processor memory. Planning engine 3 maintains and accesses a plan/model 4 representing, for example a manufacturing environment or supply chain which planning engine 3 is used to plan. Planning engine 3 has access to data 5 stored in

a fixed storage device. Data 5 can be used to initiate planning engine 3 and plan/model 4. However, during operation, planning engine 3 and plan/model 4 are typically maintained in processor memory for speed and efficiency. Planning engine 3 can provide information to a user via display 6 and can receive input data from input devices 7. In the embodiment of FIGURE 1, planning engine 3 can execute the method of the current invention. Plan/model 4 and data 5 can comprise the problem space upon which planning engine 3 operates. Planning engine 3 can execute the subsearch process as depicted in FIGURE 2.

FIGURE 2 is a block diagram of one embodiment of a subsearch process for high level controlled searching through a problem space according to the present invention.

Searching through the problem space can be achieved through the use of multiple subsearch processes 10, each comprising: a goal function 12, a termination criteria 14, a problem subset 16, a problem selection algorithm 18, a move subset 20, possible lower subsearches 22, and a script 24. Subsearch process 10 is operable to modify the model/plan 4. Problem subset 16 can include a set of problems, where, for example, a problem is an identified flaw, constraint, infeasibility, or suboptimal condition in the current search state. Problem selection algorithm 18 can include a method of selecting the next problem from the problem subset 16 to attack. Move subset 20 can include operations or moves that are operable to transform the current search state into a better one. Script 24 is operable to specify when to execute lower level subsearches 22 or a move from move subset 20. Higher level subsearch 30 of FIGURE 2 illustrates that a plurality of subsearch processes 10 can be nested.

In operation, typically, an overall search could be modeled as a subsearch process 10 having neither a move

subset 20 (or moves of its own) nor a higher level subsearch 30. Such an overall search could delegate the moves to lower level subsearches 22 and order the lower level subsearches 22 in a beneficial manner. Additionally, the present invention contemplates a scenario where lower level subsearches 22 can be viewed as "high level moves." Individual moves of move subset 20 and lower level subsearches 22 can then be ordered together in the same script 24.

One embodiment of the present invention allows for the combination of automated and manual searching mechanisms. Subsearch processes can be useful, for example, in manufacturing planning and scheduling applications, where user-intervention is critical. One such example can be the planning of overtime shifts. For instance, a human user (rather than the scheduling system) will often be in charge of planning overtime shifts. It is possible for a subsearch mechanism to take the user to a plan state in which orders are all delivered "on time" but under infeasible machine capacity conditions- thus necessitating further subsearching. The subsearch process could include the components of TABLE 1.

TABLE 1

a goal function that specifies on time deliveries;

a termination criteria of "no more late delivery problems";

a problem subset consisting of all problems whose resolution will not cause late delivery, this problem subset can exclude problems such as "machine capacity exceeded" or "material started before current time";

a problem selection algorithm which rates problems by latency and randomly selects a problem based upon the rating;

a move subset of search space moves which are operable to avoid or eliminate late delivery problems; and

no lower level subsearches.

For example, with respect to the move subset, a move such as "move task back in time" could help resolve late delivery problems, but a move such as "move task forward in time" could sometimes cause delivery problems.

In operation, a user could run this subsearch process, check machine capacities, and perhaps run some other subsearches which might resolve machine capacities. The user could then rerun this subsearch, schedule any necessary overtime shifts, and then run other subsearches to achieve a good plan under that overtime schedule. It is a technical advantage that the subsearch mechanism can provide users with several reasoning tools that are unavailable with other search mechanisms. It is another technical advantage of the present invention that searches with automated and manual features can be combined.

A further aspect of the present invention can provide for an entirely automated search mechanism. For example, in an automated embodiment, a script could comprise the flow of TABLE 2.

TABLE 2

```
first run subsearch #1
then run subsearch #2
then run subsearch #3.
```

Such an embodiment can implement a classic divide-and-conquer search strategy for dividing a slow-to-solve search into a series of fast-to-solve subsearches.

Such scripts can be implemented with a number of characteristics. One characteristic is that a script can comprise a logical flow as shown in TABLE 3.

TABLE 3

```
loop until termination condition is met, doing:
{
 if any machine's capacity in any given week
   exceeds 120%,run overtime_shift_subsearch
 else
 {
  select a subsearch randomly, weighted by priority
  numbers, and run the search for one problem
  resolution
 }
}
```

Another characteristic of a script is that it can direct a subsearch process to run a series of subsearches and then terminate. In such an embodiment, each subsearch can loop and attack (or apply moves from an associated move subset to) problems from an associated problem subset until the subsearch's termination criteria is met.

Further, a script can operate to direct a subsearch to run a series of subsearches repeatedly. The subsearch process could loop back and rerun the subsearches repeatedly until each subsearch has met its associated goal function, until a termination criteria is met, or upon manual termination.

Another characteristic is a script can operate to direct a subsearch process to run a series of subsearches repeatedly. The subsearch process can loop and look for a subsearch with an associated goal function that has not ben met and that has resolvable problems. If no such subsearch is found, the subsearch process can terminate. Otherwise, the subsearch process can run a single subsearch to attack one problem. Such an embodiment could terminate upon timeout or manual intervention.

A script can also operate to direct a subsearch process to loop until timeout or manual termination. During each loop, the subsearch process can select subsearches randomly weighted by a measure of how much work it will take to resolve problems associated with each subsearches. Each selected subsearch can be allowed to attack one of its associated problems.

A further aspect of a script is that it can offer one or more "hook" points for running lower level subsearches. For example, in pseudocode, the script could include the flow of TABLE 4.

TABLE 4

```
loop
{
  execute a designated "before search" subsearch.
  for heat = max_annealing_heat to
     min_annealing_heat by annealing_steps
  {
    resolve one of our own problems using our own
      moves.
    execute a designated "after resolve" subsearch.
     terminate if timeout or manual termination.
  }
  execute a designated "after search" subsearch.
}
```

Alternatively, the annealing loop in the embodiment could be replaced with a user-supplied termination criteria.

A number of high level computer languages can be appropriate for expressing scripts. However, it can be desirable to integrate the language with the search engine. In such an embodiment, greater search control can be achieved as the amount of accessible state information is expanded.

A subsearch's subset of moves can contribute in several ways. For example, the subset can speed the search by focusing away from undesirable states (the problems the subsearch is working to eliminate). Additionally, a subsearch can help a user understand what has happened between subsearches. For example, in a subsearch which eliminates machine capacity problems, the user could see what happens if work is delayed (but never expedited). Alternatively, by changing the allowed moves, a user could see what happens if work is expedited (but never delayed). Bundling a subset of problems to work on and a subset of moves to do in that work gives the user a high level of control and insight into the overall search.

There are many possible algorithms for selecting the next problem to attack from among a subsearch's problem subset. While various specific selection algorithms can be used, the aspect of placing a problem selection algorithm in each subsearch can provide control over the overall search. For example, a top level subsearch could be organized as a sequence of three lower level subsearches, where the first subsearch deals with "big" problems, the second subsearch deals with "medium" problems, and the third subsearch deals with "small problems." "Problems" could mean any problem, a specific type of problem, problems at a specific point in the domain (such as a highly utilized factory machine). This organization can enable faster searching and be useful in situations that require manual intervention. For example, a user can solve

the big problems, adjust the overtime schedule, and then re-execute the process to go back and solve the big problems the schedule adjustments created.

A further embodiment of the present invention can find an acceptable plan among various alternate plans. Alternate plans can arise in various cases such as when a given item can be assembled using any of several operations (each of which may process different raw parts). In pseudocode, the script of such an embodiment could include the flow of TABLE 5.

TABLE 5

```
for {the various alternate plans to try}
{ resolve(problem);
  run propagation_subsearch;
  run evaluation_subsearch;
  if evaluation_subsearch target achieved then quit;
  run cleanup_subsearch;
}
```

For example, such an embodiment can be useful in exploring alternate operations with the goal of assembling an item in a certain quantity. The propagation_subsearch can model a plan for a system to build that quantity of item and solve upstream problems that might occur based on that plan (such as the need to assemble items that go into this particular assembly operation). The evaluation_subsearch can evaluate the resultant plan, after running the propagation_subsearch, to ensure the resultant plan indeed builds the desired quantity. If so, the loop could be exited. Otherwise, the cleanup_subsearch could reset the plan so that the next alternate can be explored. For example, if propagation_subsearch built items upstream but fell short of the desired quantity, cleanup_subsearch can drive these quantity shortages downstream so that nothing is built downstream. Thus, the 'problem' can be restored, to be solved by the next alternate plan. By executing such

scripts at various levels of the bill of materials, a user can achieve sophisticated search through combinations of alternate plans.

Another embodiment of the present invention can provide a general method of finding the best plan among various alternate plans. Such an embodiment can comprise a similar subsearch process as the previously discussed embodiment. However, rather than finding the first acceptable plan, the script can direct the subsearch process to explore all alternate plans and keep the best alternate. In pseudocode, the script of such an embodiment script comprises the flow as shown in TABLE 6.

TABLE 6

```
for {the various alternate plans to try}
{ set undo point;
  resolve(problem);
  run propagation_subsearch;
  run evaluation_subsearch;
  if evaluation_subsearch target achieved then quit;
  else if goal value of evaluation_subsearch best
    ever then remember this alternate plan;
  undo;
}
set plan to the best alternate
resolve(problem);
run propagation_subsearch;
```

With such an embodiment, an "undo point" can be set which allows the plan to be restored. This embodiment can run similar subsearches as executed in the immediately prior described embodiment and store a particular alternate if the alternate is the best alternate of those explored so far. The evaluation_subsearch target can narrow a search to a reasonable number of alternates or it could just return 'false' to have all alternates searched. After the search through alternates, the plan can be set to the best alternate encountered.

Although the present invention has been described in detail, it should be understood that various changes, substitutions and alterations can be made thereto without departing from the spirit and scope of the invention as defined by the appended claims.

CLAIMS

1. A computer implemented system for executing a search through a problem space, comprising:

a search engine for controlling the execution of a number of subsearch processes; and

a plurality of subsearch processes, each comprising the following: goal data that specifies a goal of said search; termination data that specifies a termination criterium for said search; problem data that specifies problems encountered during said search; a problem selection process for selecting an ordering of said problems to be solved; a move process that executes moves, each of which transform a current search state into an improved search state; and a script process that determines whether said search shall be advanced by said move process or by another of said subsearch processes.

2. The system of Claim 1, wherein said move process of a subsearch process identifies at least one of its moves as another of said subsearch processes.

3. The system of Claim 1, further comprising a user interface in communication with said engine, said user interface operable to provide input from a user to said engine to specify subsearches to be executed.

4. The system of Claim 1, wherein said script is operable to provide a fully automated search.

5. The system of Claim 1, wherein said script of a subsearch is operable to call a move of that subsearch from another subsearch.

6. The system of Claim 1, wherein said subsearches process comprise at least the following: a propagation subsearch that provides results of a subsearch and an evaluation subsearch that evaluates said results.

7. A computer implemented system for modifying a plan modeled as part of an enterprise model, comprising:

a planning engine for controlling the execution of a number of subsearch processes; and

a plurality of subsearch processes, each comprising the following: goal data that specifies a goal of said plan; termination data that specifies a termination criterium; problem data that specifies problems encountered in obtaining said goal; a problem selection process for selecting an ordering of said problems to be solved; a move process that transforms a current search state into an improved search state; and a script process that determines whether said search shall be advanced by said move process or by another of said subsearch processes.

8. The system of Claim 7, wherein said move process of a subsearch process identifies at least one of its moves as another of said subsearch processes.

9. The system of Claim 7, further comprising a user interface in communication with said engine, said user interface operable to provide input from a user to said engine to specify subsearches to be executed.

10. The system of Claim 7, wherein said script is operable to provide a fully automated search.

11. The system of Claim 7, wherein said script of a subsearch is operable to call a move of that subsearch from another subsearch.

12. The system of Claim 7, wherein said subsearches process comprise at least the following: a propagation subsearch that provide plans and an evaluation subsearch that evaluates each of said plans.

13. The system of Claim 7, wherein said problem data represents constraints of said enterprise as modeled by said model.

14. A method of searching through a problem space, comprising the steps of:

using a search engine to control the execution of a number of subsearch processes; and

sharing searching tasks among a number of subsearch processes, each comprising the following: goal data that specifies a goal of said search; termination data that specifies a termination criterium; problem data that specifies problems encountered in obtaining said goal; a problem selection process for selecting an ordering of said problems to be solved; a move process that transforms a current search state into an improved search state; and a script process that determines whether said search shall be advanced by said move process or by another of said subsearch processes.

15. The method of Claim 14, wherein said move process of a subsearch process identifies at least one of its moves as another of said subsearch processes.

16. The method of Claim 14, further comprising the step of receiving input via a user interface in communication with said engine, said input specifying subsearches to be executed.

17. The method of Claim 14, wherein said script is operable to provide a fully automated search.

18. The method of Claim 14, wherein said script of a subsearch is operable to call a move of that subsearch from another subsearch.

19. The method of Claim 14, wherein said subsearches process comprise at least the following: a propagation subsearch that provides results of a subsearch and an evaluation subsearch that evaluates said results.

20. The method of Claim 14, wherein said search represents construction of a plan modeled as part of an enterprise model and wherein said problem space is said enterprise model.

FIG. 1

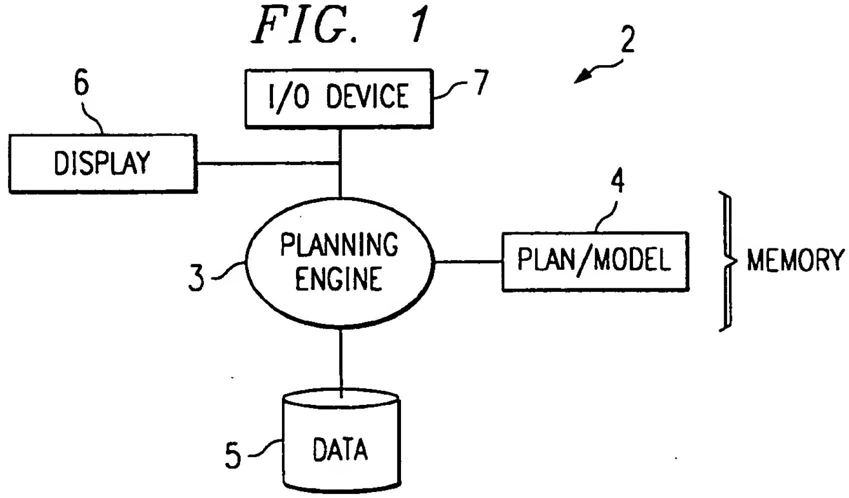
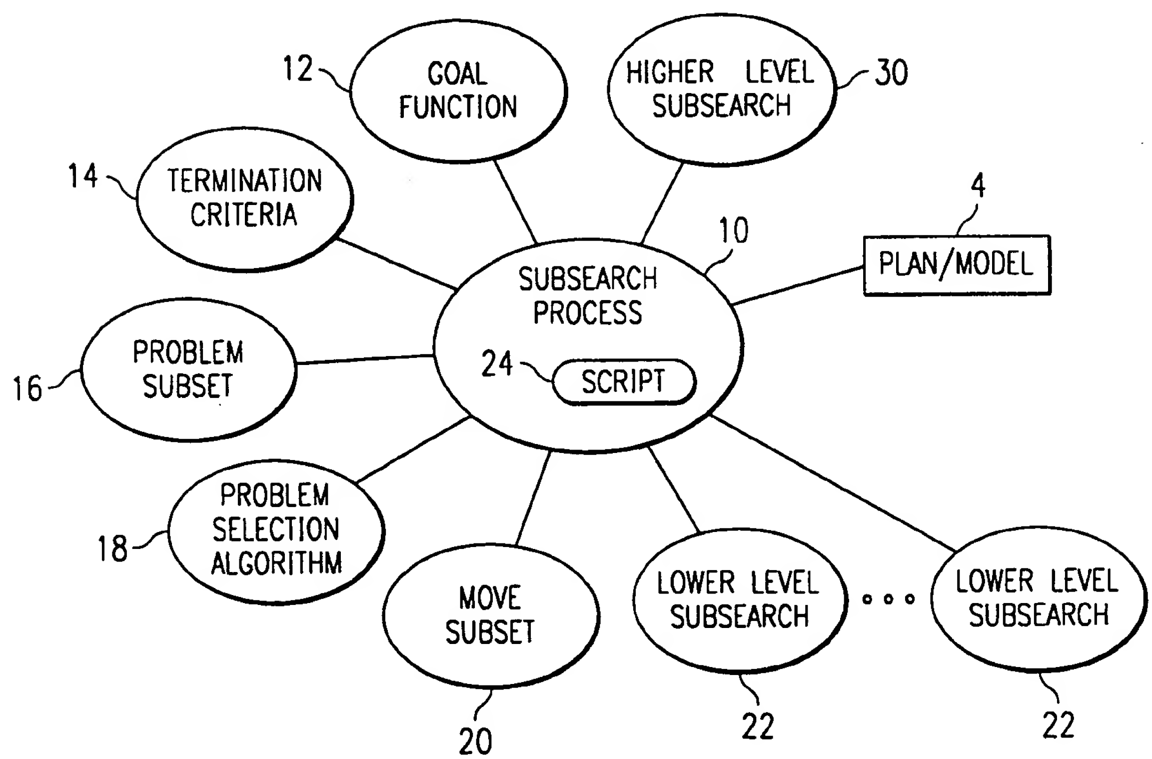

2
6
I/O DEVICE
7
DISPLAY
4
PLANNING ENGINE
PLAN/MODEL
MEMORY
3
DATA
5

FIG. 2


12
GOAL FUNCTION
HIGHER LEVEL SUBSEARCH
30
14
TERMINATION CRITERIA
4
10
PLAN/MODEL
SUBSEARCH PROCESS
PROBLEM SUBSET
24
SCRIPT
16
PROBLEM SELECTION ALGORITHM
18
MOVE SUBSET
LOWER LEVEL SUBSEARCH
LOWER LEVEL SUBSEARCH
20
22
22

# INTERNATIONAL SEARCH REPORT

| International Application No |
|---|
| PCT/US 99/11926 |

**A. CLASSIFICATION OF SUBJECT MATTER**
IPC 6 G06F17/60

According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

| Category ° | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| X | GB 2 302 425 A (I2 TECHNOLOGIES INC) 15 January 1997 (1997-01-15) page 4, line 5 - page 5, line 4 page 7, line 14 - page 11, line 34 page 18, line 4 - page 19, line 7 --- | 1-20 |
| X | US 5 737 728 A (COLLINS JOHN E ET AL) 7 April 1998 (1998-04-07) column 7, line 55 - column 9, line 12 --- | 1-3,7-9, 13-16 |
| A | EP 0 425 404 A (IBM) 2 May 1991 (1991-05-02) abstract --- | 1-20 |
| | -/-- | |

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
|---|---|
| 10 September 1999 | 16/09/1999 |
| Name and mailing address of the ISA<br>European Patent Office, P.B. 5818 Patentlaan 2<br>NL - 2280 HV Rijswijk<br>Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,<br>Fax: (+31-70) 340-3016 | Authorized officer<br><br>Bowler, A |

1

## INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 99/11926

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

| Category * | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| A | FOX M S ET AL: "ISIS - A KNOWLEDGE-BASED SYSTEM FOR FACTORY SCHEDULING" EXPERT SYSTEMS, vol. 1, no. 1, 1 July 1984 (1984-07-01), pages 25-49, XP000748117 ISSN: 0266-4720 page 38-41 ----- | 1-20 |

1

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/US 99/11926

| Patent document cited in search report | Publication date | Patent family member(s) | Publication date |
|---|---|---|---|
| GB 2302425 A | 15-01-1997 | US 5845258 A | 01-12-1998 |
| | | AT 175044 T | 15-01-1999 |
| | | AU 5599296 A | 02-01-1997 |
| | | BR 9608917 A | 02-03-1999 |
| | | CA 2176532 A | 17-12-1996 |
| | | DE 69601207 D | 04-02-1999 |
| | | DE 69601207 T | 20-05-1999 |
| | | EP 0776508 A | 04-06-1997 |
| | | ES 2128173 T | 01-05-1999 |
| | | JP 9006618 A | 10-01-1997 |
| | | WO 9700490 A | 03-01-1997 |
| US 5737728 A | 07-04-1998 | US 5467268 A | 14-11-1995 |
| | | CA 2141171 A | 26-08-1995 |
| | | EP 0669586 A | 30-08-1995 |
| | | IL 112425 A | 04-01-1998 |
| | | JP 7262273 A | 13-10-1995 |
| EP 0425404 A | 02-05-1991 | US 5099431 A | 24-03-1992 |
| | | JP 2529453 B | 28-08-1996 |
| | | JP 3136751 A | 11-06-1991 |